

Практическое занятие №13

Наименование практического занятия: составление программ с матрицами в IDE PyCharm Community.

Цели практического занятия: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с использованием с матриц в IDE PyCharm Community.

Вариант 9

Постановка задачи №1:

В матрице элементы второго столбца заменить элементами из одномерного динамического массива соответствующей размерности.

Код:

```
import numpy as np

# Создаем пример матрицы
matrix = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])

replacement_column = np.array([10, 11, 12])

# Заменяем второй столбец в матрице на значения из replacement_column
matrix[:, 1] = replacement_column

print(matrix)
```

Протокол выполнения задачи:

1. Импортируем библиотеку NumPy под псевдонимом np.
2. Создаем матрицу matrix размером 3x3, содержащую элементы от 1 до 9.
3. Создаем одномерный массив replacement_array с элементами 10, 11 и 12, который будет использован для замены.
4. В цикле по матрице matrix с помощью функции enumerate перебираем индексы и строки матрицы.
5. Для каждой строки матрицы, если индекс не равен 1 (второй столбец), строку оставляем без изменений. Если индекс равен 1 (второй столбец), заменяем строку значениями из одномерного массива replacement_array.
6. Сохраняем результаты в новую матрицу new_matrix.
7. Выводим на экран новую матрицу new_matrix.

Результат работы программы:

```
[[ 1 2 3 ]  
[ 10 11 12 ]  
[ 7 8 9 ]]
```

Постановка задачи №2:

В матрице найти среднее арифметическое положительных элементов, кратных 3.

Код:

```
def average_positive_multiples_of_3(matrix):  
    # Создаем список положительных элементов, кратных 3, с использованием  
    # спискового включения  
    multiples_of_3 = [elem for row in matrix for elem in row if elem > 0 and elem  
    % 3 == 0]  
  
    # Вычисляем сумму всех таких элементов  
    total = sum(multiples_of_3)  
  
    # Вычисляем количество элементов в списке  
    count = len(multiples_of_3)  
  
    # Если элементов нет, возвращаем 0, чтобы избежать деления на 0  
    if count == 0:  
        return 0  
  
    # Возвращаем среднее арифметическое положительных элементов, кратных 3  
    return total / count  
  
# Пример матрицы  
matrix = [  
    [3, -6, 9],  
    [12, 5, 18],  
    [-3, 6, 7]  
]  
  
result = average_positive_multiples_of_3(matrix)  
print("Среднее арифметическое положительных элементов, кратных 3, в матрице:",  
result)
```

Протокол выполнения программы:

1. Определяется функция `average_positive_multiples_of_3(matrix)`, которая принимает матрицу в качестве входного параметра.
2. Создается список `multiples_of_3` с помощью спискового включения, где происходит

итерация по строкам и элементам матрицы, и добавляются только положительные элементы, кратные 3.

3. Вычисляется сумма всех положительных элементов, кратных 3, с помощью функции `sum(multiples_of_3)`.

4. Определяется количество элементов в списке `multiples_of_3` с помощью `len(multiples_of_3)`.

5. Если количество элементов равно 0, то функция возвращает 0, чтобы избежать деления на 0.

6. Возвращается среднее арифметическое положительных элементов, кратных 3, как отношение суммы к количеству элементов.

7. Задается пример матрицы `matrix`.

8. Вызывается функция `average_positive_multiples_of_3(matrix)` с матрицей в качестве аргумента и результат присваивается переменной `result`.

9. Выводится сообщение с результатом выполнения функции: "Среднее арифметическое положительных элементов, кратных 3, в матрице: {result}".

Результат работы программы:

Среднее арифметическое положительных элементов, кратных 3, в матрице: 9

Вывод:

Я закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрел навыки составления программ с использованием с матриц в IDE PyCharm Community.